

論文紹介

Why Does Code Review Work for Open Source Software Communities?

なぜコードレビューは OSSコミュニティのために機能するのか？

コードレビュープロセスの人間的・社会的側面を見つけ出し
コードレビューを改善するアイデアを提案する

著者

Adam Alami, Marisa Leavitt Cohn, Andrzej Wasowski
IT University of Copenhagen Denmark

紹介者：堀口日向

1. 概要

4つのOSSコミュニティの
開発者へのインタビュー



Robot Operating Systemへの
コードレビューの導入について
話し合うミーティング



音声を文字に書き起こし
定性的分析



分かったことを使い
プロジェクトにおけるコードレビューを
より良いものにする方法を提案

2. 背景 (1/2)

コードレビューはソフトウェア業界で広く利用

コードレビューの効果

ソースコードの品質保証 バグ発生の頻度を下げる コーディング規約を守る

特にOSSプロジェクトにおいては
誰でもプロジェクトに貢献できれば、混乱が生じる可能性がある
非常に重要視

2. 背景 (2/2)

OSSコミュニティのコードレビューの本質的な問題

レビュー者の感情が含まれたり、公開された状態でやり取りするので
意見が衝突する可能性

しかし、OSSコミュニティは高品質なソフトウェアを
生み出すことができることを実証した（例：Linux Kernel, PostgreSQL, etc.）



Q. なぜコードレビューはOSSコミュニティでうまく機能するのか？

目的

他のOSSコミュニティや、クローズドソース環境において
コードレビューをうまく実行する方法を提案

3. 方法 (1/2)

参加者の経験や動機を明らかにするのに適した、定性的研究方法を用いる

1. データ収集 (会話の記録)

コードレビューを使用している
OSSコミュニティの計21人の開発者へ
インタビュー

Apache Allura (3人) , CKAN (4人) ,
FOSSASIA (6人) , Linux Kernel (8人)



主に既存のコードレビューでの
経験についての情報が得られる

Robot Operating Systemへの
コードレビューの導入について
話し合うミーティング計3回

10~16人の参加者



コードレビューに対する期待、願い、
懸念についての情報が得られる

3. 方法（2/2）

2. 定性的分析

インタビューやミーティングで集まった会話の文章をオープンコーディングで分析

手順

1. 会話の文章を1文ずつ切り分ける
2. その1文が何を表しているのか、何が起きているのか、何を伝えようとしているのかを考え、ラベル付けする
3. ラベル同士をカテゴリ別にまとめて、テーマを見つける

見つけたテーマは

Q. なぜコードレビューはOSSコミュニティでうまく機能するのか
の質問に答えるための暗黙のトピック

4. 見つけたテーマについて (1/3)

6つのテーマを識別した

1. 拒否 (Rejection)

コントリビュータはコードレビューにおいて頻繁に拒否される
コミュニティはそれがコードレビューの本質であると信じているので
否定的なフィードバックを減らすこともなくすこともしない

2. 反復的な改善 (Iterative improvement)

コードレビューの反復的な改善サイクルは、否定的なフィードバックを
コントリビュータによる学習および技術的成長のための前向きな機会に変える
フィードバックを受け取ることが参加の理由になることさえある

4. 見つけたテーマについて (2/3)

3. 情熱 (Passion)

コントリビュータを動機づけ、コードレビューに努力を注ぎ
高品質を提供し、そして拒否に対して回復力がある

4. 思いやり (Caring)

思いやりは本研究の疑問の答えに近い
レビュー者の否定的なフィードバックを制御（暴言の抑止）するのに役立つ

4. 見つけたテーマについて (3/3)

5. 内的動機 (Intrinsic motivation)

利他主義、楽しさは、本研究の疑問にとって重要な内的動機
オープンソースのレビュー担当者は（有償であっても）事実上のボランティアで
自分の興味に従ってレビュータスクを選択することができる

6. 外的動機 (Extrinsic motivation)

成功の共有、労働市場に対する自分の地位の向上、新しい技術の学習の機会は
本研究の疑問とコードレビューの背後にある主な外因的動機である
これらはコードレビューを改善するために使用できる非金銭的な動機

5. 提案（1/3）

コードレビューをより良いものにするためにプロジェクト管理者がすべきこと

見つけたテーマから分かること	プロジェクト管理者への提案
コードレビューで貢献が拒否されることは コードレビューを重視する環境であれば当たり前の ことであり、受け入れるべき	・フィードバックを伝える方法と それを建設的に解釈するためのガイドラインの提供 ・拒否を扱うことを新しいエンジニアのトレーニングに 含める（拒否は失敗ではないと教える） ・エンジニアがコードレビューのプラスの価値を 理解するための社内カウンセリング、メンタリング、 コーチング機能確立する
反復的な改善は、完璧を目指すための努力であり プログラミング作業の重要な方法	・知識の共有と学習の機会としてのコードレビューを 促進する ・学ぶこと、チームと個人の両方で優秀な開発者になる ことを目指して努力することに対して報酬を与える ・レビューアがレビューしたいものを選択できるように

5. 提案（2/3）

コードレビューをより良いものにするためにプロジェクト管理者がすべきこと

見つけたテーマから分かること	プロジェクト管理者への提案
情熱は、プロジェクトチームやコミュニティで育まれるべき	・可能であれば、(特にコードレビューに対して)情熱的な開発者を募集する。ソフトウェアエンジニアをリードするために情熱的な人をレビュアーに任命する ・開発者の情熱を浪費するタスクを排除 ・開発者がタスクに対して情熱を持てるようにタスクについての自己決定ができるようにする
思いやりは、コードレビューに携わるプロジェクトチームに良い働きをする	・メンターとコントリビュータの関係を確立することによってコントリビュータへの思いやりを育む関係を築くことが、思いやりの鍵

5. 提案 (3/3)

コードレビューをより良いものにするためにプロジェクト管理者がすべきこと

見つけたテーマから分かること	プロジェクト管理者への提案
エンジニアの内的動機、利他主義、そして楽しみを育む	内的動機を外部から誘発するのは困難 今後の研究課題
多くの非金銭的な外的動機が、コードレビューをより効果的にするためにOSSのコントリビュータを駆り立てる	• 金銭以外の精神的な報酬を無視せずコラボレーションから報酬が得られるようなチーム内関係を築く • プロジェクトが成功したら、コミュニティメンバーを惹きつけるためにそれを利用する • 実力主義の階層構造の構築 • 評判と地位を獲得するためにポイントシステムの使用 • 個人およびグループの業績を定期的に認め、昇進、会議の話し合いなどを使ってそれに報いる • 開発者にとって重要な、学習および開発の機会としてコードレビューを使う

6. まとめ

Q. なぜコードレビューがOSSコミュニティでうまく機能するのか？

A. 拒否、反復的な改善、情熱、思いやり、内的動機、外的動機があるから

本研究での提案は

- ・プロジェクトやコミュニティの管理者にとって有益なインスピレーション
- ・クローズド/オープンソースプロジェクトの両方に活用可能でコードレビューをよりよくすることにつながる